A Method for Representing Search Results in Three Dimensions

Michael H. Miller
School of Health Information Science
University of Victoria
Victoria BC
V8W 3P5
email: mikem@uvic.ca

This paper presents a new method for representing results of an information retrieval search in a three dimensional environment. Aside from the fact that users find 3-D interfaces visually appealing, there are strong practical reasons for developing 3-D representations of search results. Traditional information retrieval systems present results in ordered lists which are difficult to browse, and exclude useful information. The current method employs a multivariate statistical method called Local Latent Semantic Indexing (LLSI) to create meaningful local dimensions in which to view search results. A prototype Internet-ready system is described which utilizes Virtual Reality Modeling Language (VRML) to display search results. Preliminary tests of this system with a small collection of MEDLINE articles are very encouraging.

INTRODUCTION

Despite advances in information technology for searching bibliographic databases, physicians and other health professionals are not keeping pace with the biomedical literature.[1] One reason for this fact is that many people find it extremely difficult to find relevant information with current information retrieval (IR) systems.[2] Effective on-line searches require users to be well acquainted with a controlled vocabulary such as the Medical Subject Headings (MeSH); few have mastered this skill.[3] Another reason is that much of the research conducted in the area of information retrieval focuses on "system" issues and largely ignores the user component of an information search.

Aside from the fact that users find 3-D interfaces very appealing, there are sound reasons for developing visual retrieval environments. Current retrieval systems are not well suited for browsing or suggesting alternate search methods. Today's systems present the user with a limited set of documents – those judged most relevant by the system – and the retrieved documents are presented in a sequential list with little or no indication of the system's evaluation of each document. A typical MEDLINE search produces a list of hundreds of articles that are potentially relevant to a users' query. It is often difficult for the user to tell how far down the list she should look for relevant articles.

Some systems include numerical "confidence rankings" (e.g., SMART[4], INQUERY[5]) of the similarity between the document and the query, but the salience of these values is difficult to easily comprehend. Moreover, these confidence ratings do not allow one to gauge the similarity of a given document in the list to another. Furthermore, most systems offer no information about documents that the system rejected. If the user does decide to reformulate the query from scratch, he or she does so without any guidance from the system.

A subtler issue arises from condensing similarity to a single value. By doing this, a retrieval system artificially imposes a one dimensional measure of "likeness". In fact, documents may be similar (or different) in a variety of ways. For instance, articles that match the topic, *side effects of cyclosporine,* may differ on the type of subjects used (animal or human), the experimental design (randomized clinical trial or case study), and the particular side effects of interest. From a users point of view, one would like to see the similarity between documents represented in several dimensions.

DIFFICULTIES INVOLVED IN CREATING A 3-D SEARCH SPACE

The difficult part of creating a 3-D view of search results is devising a way of representing the maximum number of relevant documents along a very small number of meaningful dimensions. Previous efforts in representing document spaces visually are limited by synonymy problems or lack of meaningful local axes. Some systems based on the VSM take two or three strong vectors from the global document space, along which document clusters

1091—8280/97/$5.00

form.[6,7] With this approach, a document's strength of association with a concept is indicated by its position in the space. A critical problem with this approach is that the technique is very sensitive to the choice of vocabulary. For instance if the user phrases the previous example query as *adverse effects and method of action of cyclosporine* instead of *side effects and pharmacodynamics of cyclosporine*, the resulting spaces may have a very low overlap of documents. A more recent approach overcomes this synonymy problem by utilizing a probabilistic model (INQUERY) to account for associations between terms.[8] However, to reduce the number of dimensions to three, the system uses a complex method called spring embedding which decouples the axes from any special meaning, making it impossible to represent the similarity between documents spatially. The method presented in this paper uses an extension of the basic LSI model called Local LSI (LLSI) which handles synonymous term usage and creates meaningful local dimensions.

CREATING A VISUAL SEARCH SPACE WITH LOCAL LSI

Latent Semantic Indexing (LSI) is an extension of the vector space model[9] that derives virtual constructs from the occurrences of words within and between documents.[10] This is accomplished by simultaneously modeling all of the interrelationships among words in each document and the collection as a whole. The model is constructed using a Singular Value Decomposition (SVD), a dimension reduction technique related to factor analysis. For the purposes of information retrieval, the SVD is a technique for deriving a reduced set of orthogonal indexing constructs from the original term-document matrix. Loosely speaking, the constructs can be thought of as concepts that represent the extracted common themes of many different documents. These virtual concepts have no surface level meaning; one might say that they represent the latent semantic structure of documents that make up the collection.

In the LSI model each document and term is characterized by a vector of weights that indicates the strength of its association with each of the constructs. Since the number of concepts is much smaller than the number of unique terms (typically 100-300 constructs are retained), words are not independent in the LSI model. In fact, it is possible for documents or queries with slightly different term usage to be mapped to the same concept. For instance, with an LSI index, the query *high blood pressure* may be considered very similar to articles that contain only the synonym *hypertension.* This is because documents containing the two phrases are likely to be very similar in their content. A few studies using LSI with small subsets of MEDLINE found 10-30% increases in retrieval performance over the VSM.[11,12]

The Local LSI (LLSI) method involves creating a specific set of constructs based on a small section of the LSI document space (the index). In brief, this involves deriving a new set of local constructs for each iteration of feedback via a second SVD of a subsection of the global document space, and then folding all of the other documents into this local space.[13] Figure 1 illustrates the usefulness of this procedure. Two hundred document objects are shown in this 3 dimensional LLSI space defined by some of the top ranked documents returned in response to the query *treatment of multiple myeloma.* To begin the process, the user judges a small number (e.g., 10) of articles presented by an LSI search engine. These 10 articles are then used to define three local dimensions. The location of all other documents in the local space can then be quickly calculated.

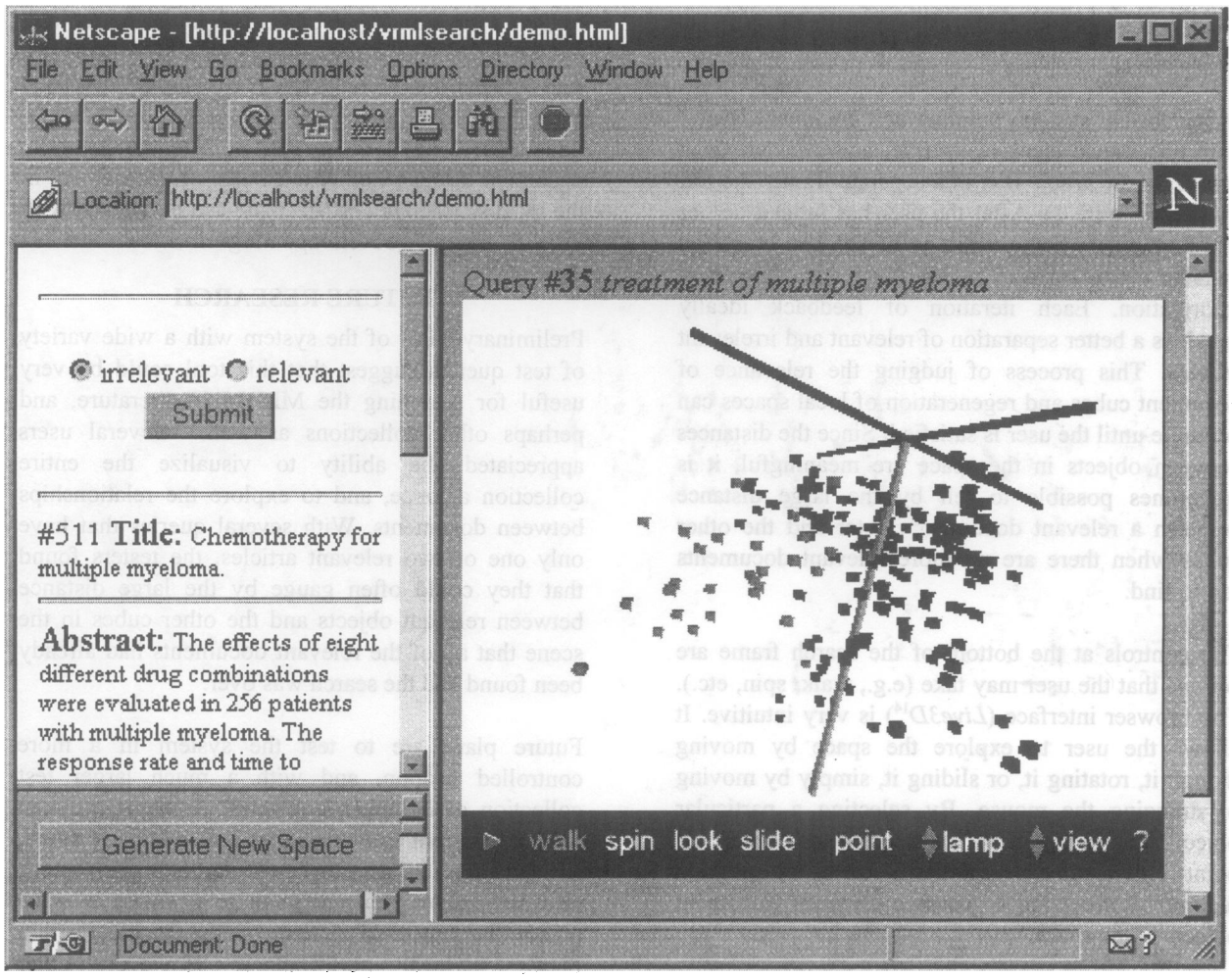

Figure 1. An example local space based on the query, *treatment of multiple myeloma*. Note that the majority of the relevant documents (light gray cubes) are very close to the sole relevant document found so far (the sphere in the lower left of the figure).

The solid lines in the graph in figure 1 represent the three local constructs derived in the local SVD. Each small cube represents a document. Relevant documents tend to be located in a cluster at some distance from the origin of the graph. Because they have little or nothing in common with the local constructs, irrelevant articles tend to cluster around the origin. In this example, only the 200 documents furthest from the origin are displayed. In theory, the entire collection can be viewed in the space. However, since relevant documents should have a strong association with one or more of the local concept dimensions, they are invariably found far from the origin. Therefore, little, if any, information is lost by omitting the bulk of the documents from the scene.

The small lightly shaded sphere at the lower left of the graph represents the one relevant article found so far in this example. The lightly shaded cubes represent the relevant documents for this particular query which the user has not yet judged, and the darker cubes represent non-relevant documents that the user has not yet judged. Obviously, in an ad-hoc search the location of relevant articles in the space is not known—they are color coded here only for illustrative purposes. The majority of articles relevant to the query, *treatment of multiple myeloma* are located in a cluster near the only known relevant article found so far in the example. Using the heuristic that relevant articles tend to cluster together away from the origin, and near other relevant articles, a user could quite quickly and accurately locate a large number of the relevant document cubes. Without feedback, both LSI and SMART (a VSM model) do relatively poorly on this particular query;

mean precision with both of these methods is below 35 percent.

In the current system, pointing at a document cluster with the mouse displays its title, and clicking on it brings up the article text in a separate frame (the left frame in Figure 1). After the user has judged one or more candidate documents, he or she can create an updated local space which includes this new information. Each iteration of feedback ideally provides a better separation of relevant and irrelevant articles. This process of judging the relevance of document cubes and regeneration of local spaces can continue until the user is satisfied. Since the distances between objects in the space are meaningful, it is sometimes possible to tell by the large distance between a relevant document cluster and the other cubes when there are no more relevant documents left to find.

The controls at the bottom of the search frame are actions that the user may take (e.g., walk, spin, etc.). The browser interface (*Live3D*[14]) is very intuitive. It allows the user to explore the space by moving though it, rotating it, or sliding it, simply by moving or dragging the mouse. By selecting a particular object, the user can rotate the entire space around this point. This feature is particularly useful for getting a feeling of the relative distance between document objects in the space.

THE TEST COLLECTION

The test collection used in preliminary trials of the system consists of 2344 articles from MEDLINE, along with 75 queries written by physicians.[15] The collection also includes a list of relevance judgments that indicate which articles are relevant to each query. The words in each title and abstract were preprocessed by removing words on a stop list, reducing the remaining words to their stem form, and weighting the word stems.[16] The resulting term/document matrix was indexed using an LSI based system developed by the author.

IMPLEMENTATION

The prototype system runs as an Internet Common Gateway Interface (CGI) application woven together by Perl scripts and C++ code. The initial search and the local spaces are computed with a C++ program. The output from the local space C++ program is used by a Perl script to create Virtual Reality Markup Language (VRML) code that defines the location, document title, and other attributes of each object in the local space. The VRML code can be viewed with Live3D, a standard Netscape Plug-in.

FUTURE RESEARCH

Preliminary trials of the system with a wide variety of test queries suggest that this tool could be very useful for searching the MEDLINE literature, and perhaps other collections as well. Several users appreciated the ability to visualize the entire collection at once, and to explore the relationships between documents. With several queries that have only one or two relevant articles, the testers found that they could often gauge by the large distance between relevant objects and the other cubes in the scene that all of the relevant documents had already been found and the search was over.

Future plans are to test the system in a more controlled fashion, and with a much larger test collection of MEDLINE articles. A major problem with the current test set is that it is relatively small, and contains an unrealistically large proportion of relevant articles. It is difficult to estimate to what extent the clustered nature of the test collection biases retrieval performance. However, other research in which LLSI has been adapted to large, unclustered collections--although not for the purpose of visualization--suggests that the method will scale up well.[17]

ACKNOWLEDGEMENTS

The author would like to thank Dr. Bill Hersh for making the MEDTEST collection available. Also thanks to Dr. David Maxwell, and Dr. Yuri Kagolovsky for trying out the system, and to Jim McDaniel for helpful comments and advice with C++ programming concerns.

REFERENCES

[1] Williamson JW, German PS, Weiss R, Skinner EA, Bowes F. Health sciences information management and continuing education of physicians. *Annals of Internal Medicine.* 1989;110:151-160.

[2] Borgman CL. Why are online catalogs hard to use? Lessons learned from information retrieval studies. *Journal of the American Society for Information Science.* 1986;37:387-400.

[3] Sewell W. & Teitelbaum S. Observations of end-user on-line searching behavior over eleven years. *Journal of the American Society for Information Science.* 1986;37:234-245.

[4] Salton G, McGill MJ. *Introduction to modern information retrieval.* New York: McGraw-Hill. 1983.

[5] Callan JP., Croft WB., and Harding S.M. The INQUERY Retrieval System. *Proceedings of the 3rd International Conference on Database and Expert Systems Applications.* 1992;78-83.

[6] Hemmje M, Kunkel C, Willett A. LyberWorld - a visualization user interface supporting fulltext retrieval in *Proceedings of SIGIR '94*, 1994:199-204.

[7] Dubin D. Document analysis for visualization. In *Proceedings of SIGIR '95.*1995;199-204.

[8] Swan RC & Allan J. Improving interactive information retrieval effectiveness with 3-D graphics. 1996. Online; Available: http://ciir.cs.umass.edu/info/psfiles/irpubs/ir-100.ps.gz

[9] see note 4.

[10] Deerwester S, Dumais ST, Furnas GW, Landauer TK, Harshman R. Indexing by latent semantic analysis. *Journal of the American Society for Information Science.* 1990; 41(6):391-407.

[11] see note 10.

[12] Miller MH. A bottom-up, concept based methodology for indexing and searching MEDLINE. In *Information Technology and Community Health Proceedings '96.* 1996:22-27.

[13] Hull D. Improving text retrieval for the routing problem using latent semantic indexing. *In Proceedings of SIGIR '94.*1994;282-289.

[14] The Live3D Virtual Reality Modeling Language (VRML) browser is freely available from Netscape Corporations at: http://cgi.netscape.com/comprod/products/navigator/live3d/index.html

[15] Haynes R, McKibbon K, Walker C, Ryan N, Fitzgerald D, Ramsden M. Online access to MEDLINE in clinical settings. *Annals of Internal Medicine.* 1990;112:78-84.

[16] Salton G & Buckley C. Term-weighting approaches in automatic text retrieval. *Information Processing and Management.* 1988;24:513-523.

[17] Shutze H, Pederson J, Hull DA. A comparison of classifiers and document representations for the routing problem. In *Proceedings of SIGIR '95.* 1995:229-237.